BAB II
LANDASAN TEORI

2.1 Dataset NSL-KDD

Network Security Layer-Knowledge Discovery in Database (NSL-KDD) merupakan set data terbaik dan menjadi titik acuan untuk menguji sistem kerja IDS. *Dataset* ini merupakan versi kelanjutan dari *dataset* KDDcup“99 dan memiliki beberapa keunggulan dibandingkan dengan *dataset* sebelumnya yang berhasil memecahkan masalah yang ada pada *dataset* KDDcup”99. NSL-KDD biasanya digunakan untuk mempelajari apakah algoritma klasifikasi dalam mendeteksi anomali pada pola lalu lintas jaringan efektif atau tidak (Dhanabal and Shantharajah, 2015).

Dataset NSL-KDD terdiri dari *record* yang dipilih dari versi sebelumnya yaitu KDD*cup*”99, terdapat beberapa keuntungan menggunakan *dataset* ini. Salah satu keuntungan yang diperoleh adalah tidak adanya *record* data yang berlebihan di dalam *train set*, sehingga *classifier* tidak akan menghasilkan hasil yang bias. *Datasets* NSL-KDD ini tidak memiliki duplikat *record* pada *tes set* yang memiliki *reduction rates* yang lebih baik dan jumlah *record* yang dipilih dari setiap *level* grup yang berbeda-beda berbanding terbalik dengan persentasi *record* di dalam *dataset* KDD yang asli (Dhanabal and Shantharajah, 2015).

Di dalam dataset NSL-KDD terdapat 5 kategori serangan (Seth, 2017). 5 Kategori serangan seperti akan dijabarkan pada Tabel 2.1 berikut:

Tabel 2.1 *Dataset* NSL-KDD

No	Tipe Kelas	NSL-KDD
1	Normal	67343
2	DoS	45927
3	*Probes*	11656
4	U2R	52
5	R2L	995
	Total	**125.973**

1. Normal, adalah suatu aktivitas yang dikategorikan sebagai normal dan dianggap bukan bentuk serangan atau ancaman kepada data informasi.
2. *Denial of Service* (DoS), adalah sebuah bentuk serangan yang membebani sumber daya pada komputer sehingga komputer yang menjadi target mengalami kerusakan dan tidak mampu untuk memproses koneksi normal bahkan berakibat pengguna tidak dapat mengakses perangkat tersebut.
3. *Probes*, serangan ini bertujuan untuk mendapatkan informasi tentang status jaringan pada komputer dengan cara melakukan pemindaian terhadap beberapa komputer dalam jaringan tersebut. Informasi ini dapat digunakan oleh penyerang untuk memetakan jaringan yang berguna dalam melakukan penyerangan berikutnya.
4. *User to Root* (U2R), adalah bentuk serangan yang berusaha untuk mendapatkan akses *root* pada komputer yang menjadi target dengan melakukan eksploitasi celah pada keamanan sistem. Serangan U2R umumnya dilakukan setelah penyerang mendapatkan akses *user* normal ke sistem.
5. *Remote to local* (R2L), adalah bentuk serangan yang bertujuan untuk mendapatkan akses sebagai pengguna sistem. R2L dapat dilakukan oleh penyerang yang memiliki akses ke sistem dan melakukan eksploitasi untuk mendapatkan akses lokal.

Terdapat beberapa kelebihan pada NSL-KDD *dataset* dibandingkan dengan *dataset* sebelumnya yaitu KDDcup”99, beberapa kelebihan tersebut diantaranya (Revathi and Malathi, 2013):

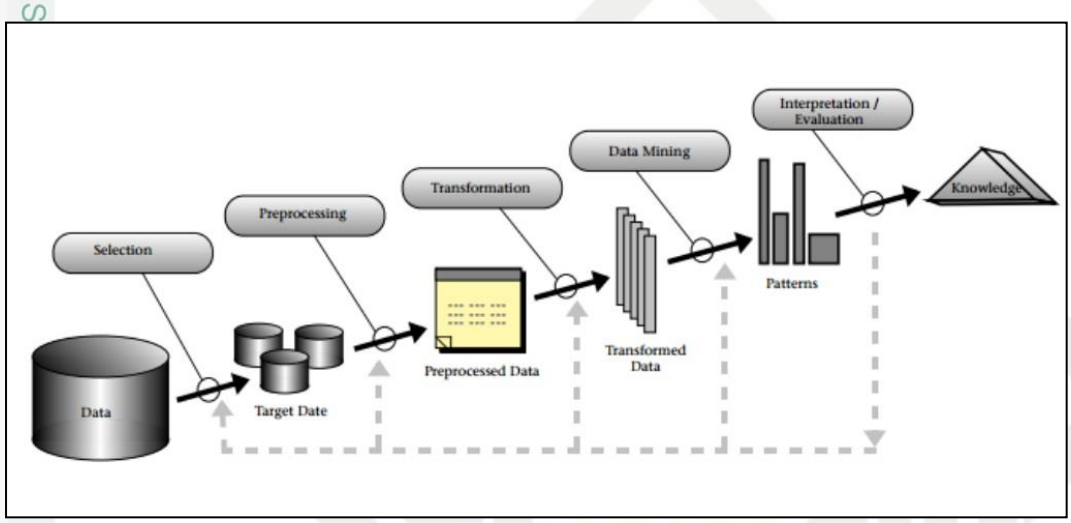
1. Pada *train set record* yang berlebihan tidak ada, sehingga *classifier* tidak akan menghasilkan bias.
2. Pada *train set* duplikat *record* tidak ada yang memiliki pengurangan nilai yang lebih baik.
3. Pada *dataset* NSL-KDD asli jumlah *record* yang dipilih berbanding terbalik dari setiap *level* grup yang berbeda.

2.2 *Knowledge Discovery in Database* (KDD)

Data mining dan *Knowledge Discovery in Database* (KDD) digunakan untuk menjelaskan proses penggalian informasi dari sebuah data yang besar, namun

kedua istilah tersebut memiliki konsep yang berbeda tetapi saling berkaitan antara satu dengan lainnya. *Data mining* adalah salah satu tahapan dari keseluruhan tahap yang ada pada KDD.

Knowledge Discovery in Database (KDD) merupakan sebuah proses untuk menemukan informasi berguna yang terdapat pada *dataset*. Informasi tersebut terdapat di dalam sebuah basis data yang berukuran besar dan sebelumnya belum diketahui namun berpotensi mempunyai informasi yang bermanfaat. (Fayyad, 1996). Proses tersebut akan ditampilkan pada Gambar 2.1 berikut:



Gambar 2.1 Tahapan KDD (Fayyad, 1996).

Adapun penjelasan dari Gambar 2.1 adalah sebagai berikut:

1. *Data Selection*
   - Data *selection* merupakan proses pengambilan data yang berhubungan dengan analisis dari basis data. Pada tahapan ini dilakukan teknik perolehan sebuah pengurangan representasi dari data dan meminimalkan hilangnya informasi data. Hal ini meliputi metode pengurangan atribut dan kompresi data.
   - Pemilihan (seleksi) data dari sekumpulan data operasional harus dilakukan sebelum tahap KDD dimulai. Data hasil seleksi yang akan digunakan untuk proses *data mining* disimpan dalam suatu berkas, terpisah dari basis data operasional.

- Pada penelitian ini metode pengurangan atribut yang digunakan adalah kombinasi metode *feature selection symmetrical uncertainty* dan *gain ratio*.

2. *Data Cleaning* (*Pre-Processing*)
   - Pemprosesan pendahuluan dan pembersihan data merupakan operasi dasar seperti dilakukannya penghapusan *noise*.
   - Sebelum proses *data mining* dapat dilaksanakan, perlu dilakukan proses *cleaning* pada data yang menjadi fokus KDD.
   - Proses *cleaning* mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data.
   - Dilakukan proses *enrichment*, yaitu proses memperkaya data yang sudah ada dengan data atau informasi lain (eksternal).
3. *Data Transformation*
   - Merupakan proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses *data mining*. Proses ini merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.
4. *Data Mining*
   - Pemilihan tugas *data mining*, pemilihan tujuan dari proses KDD misalnya klasifikasi, regresi, *clustering*, dan lain-lain.
   - Proses *data mining* yaitu proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam *data mining* sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.
   - Pada penelitian ini menggunakan metode klasifikasi *Modified K-Nearest Neighbor* (MK-NN).
5. *Evaluation*
   - Penerjemahan pola-pola yang dihasilkan dari *data mining*.

- Pola informasi yang dihasilkan dari proses *data mining* perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan.
- Tahap ini merupakan bagian dari proses KDD yang mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesa yang ada sebelumnya.

2.3 *Feature Selection*

Feature Selection merupakan teknik *pre-processing* dalam penambangan data dan pemilihan atribut yang bertujuan untuk menemukan informasi dari data yang disimpan pada *dataset* NSL-KDD (Das, Sengupta and Bhattacharyya, 2018). Pada kriteria penilaian tertentu, subset fitur yang optimal dipilih dari seluruh set fitur. *Feature Selection* digunakan untuk menghapus fitur-fitur yang tidak relevan dan berlebihan dari *dataset*, sehingga dapat meningkatkan akurasi dan mempersingkat waktu dalam melakukan klasifikasi. Metode *feature selection* dapat dikategorikan dalam dua kategori diantaranya *feature selection* berbasis *filter* dan *wrapper* (Das, Sengupta, and Bhattacharyya, 2018).

2.3.1 Berbasis *Filter*

Pada *feature selection* berbasis *filter*, semua fitur diberi nilai dan diurutkan berdasarkan kriteria tertentu. Fitur yang mendapatkan peringkat tinggi akan dipilih dan kemudian fitur dengan nilai rendah akan dihapus. Metode ini mudah beradaptasi dengan *dataset* yang sangat besar. *Feature selection* ini dilakukan dalam sekali pemilihan, setelah itu *dataset* yang dipilih sudah bisa dievaluasi pada berbagai metode klasifikasi (Kumari and Swarnkar, 2011). Pada penelitian yang akan dilakukan kali ini, metode *feature selection* berbasis *filter* yang digunakan adalah sebagai berikut:

a. Symmetrical Uncertainty

Merupakan sebuah pendekatan untuk mengukur nilai sebuah fitur dengan mengukur ketidakpastian simetris yang berhubungan dengan kelas, dan mengkompensasi bias pada metode *feature selection information gain* (Garg and Kumar, 2014).

Persamaan *Symmetrical Uncertainty* ditunjukkan pada Persamaan (**2.1**).

$$\boldsymbol{SU = 2.0\left(\frac{information\ gain}{I(D) + I(A)}\right)} \quad \textbf{(2.1)}$$

Sedangkan persamaan *information gain* adalah seperti pada (**2.2**):

$$\boldsymbol{Information\ Gain\ (y, x) = I\ (D) - I(A)} \quad \textbf{(2.2)}$$

Keterangan:

I(D) : Entropy

I(A) : Entropy antribut

Persamaan I(D) didapat dari Persamaan (**2.3**).

$$\boldsymbol{Info\ (D) = -\sum_{i} P(x_i)\ log_2\left(P(x_i)\right)} \quad \textbf{(2.3)}$$

Keterangan :

D : himpunan kasus

P(Xi) : Proporsi dari D*i* terhadap D

Persamaan I (A) didapat dari Persamaan (**2.4**).

$$\text{Info (A)} = \sum_{j=1}^{v} \frac{D_j}{D} \text{x I}\left(D_j\right) \quad \textbf{(2.4)}$$

Keterangan :

A : Atribut

v : Jumlah partisi atribut A

|Dj| : Jumlah kasus pada partisi ke j

|D| : Jumlah keseluruhan data

I (Dj) : Entropy dalam partisi

b. Gain Ratio

Gain Ratio adalah salah satu modifikasi *feature selection* yaitu *information gain* yang bertujuan untuk mengurangi bias atribut yang memiliki banyak cabang (Garg and Kumar, 2014).

Feature selection ini memiliki Persamaan seperti (**2.5**):

$$\boldsymbol{Gain\ Ratio\ (y, x) = \frac{\textbf{Information Gain}\ (y, x)}{\textbf{Split Info}(x)}} \quad \textbf{(2.5)}$$

Persamaan *Split Info* didapat dari Persamaan (**2.6**)

$$\boldsymbol{Split\ Info}\ (\mathbf{x}) = -\sum \frac{|\mathbf{Di}|}{|\mathbf{D}|}\ \mathbf{x}\ \mathbf{Log_2}\ \frac{|\mathbf{D_i}|}{|\mathbf{D}|} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (\mathbf{2.6})$$

Keterangan :

|Dj| : Jumlah kasus pada partisi ke j

|D| : Jumlah kasus dalam D

2.4 *Data Mining*

Data mining merupakan suatu cabang ilmu pengetahuan yang mempelajari metode untuk menemukan sebuah informasi yang bermanfaat dari data berskala besar. *Data mining* memanfaatkan teknik statistik, matematika, kecerdasan buatan dan *machine learning* untuk mengekstraksi dan mengambil informasi yang bermanfaat dari berbagai data berukuran besar. Hasil dari *data mining* dapat berupa sebuah kesimpulan atau informasi (Fayyad, 1996).

Data mining memiliki 4 tujuan utama, diantaranya adalah:

1. Klasifikasi (*Classification*)

   Klasifikasi bertujuan untuk mengklasifikasikan sebuah data kedalam salah satu dari kelas data standar.

2. Regresi (*Regression*)

   Regresi merupakan sebuah tahap dilakukan pemodelan hubungan antara dua variabel atau lebih.

3. Pengelompokan (*Clustering*)

   Clustering adalah metode yang bertujuan untuk mengelompokkan sejumlah data ke dalam grup. Pada setiap grup akan berisi data yang seserupa mungkin.

4. Pembelajaran Aturan Asosiasi (*Assosiation Rule Learning*)

   Pembelajaran aturan asosiasi bertujuan untuk mencari hubungan antara variabel. Sebagai contoh fakultas sains dan teknologi mengumpulkan data jurusan yang paling banyak dipilih oleh mahasiswa baru. Dengan menggunakan pembelajaran aturan asosiasi (*Assosiation Rule Learning*), fakultas sains dan teknologi dapat menentukan jurusan yang paling diminati

oleh mahasiswa baru dan menggunakan informasi ini untuk peningkatan mutu jurusan.

Teknik yang digunakan dalam *data mining* didukung oleh 3 teknologi yaitu pengumpulan data secara besar, *multiprocessor* pada komputer dan algoritma *data mining*. Adapun tugas dari *data mining* adalah:

1. Deskriptif

   Yaitu menemukan gambaran pola dari data yang diproses.

2. Prediktif

   Yaitu memprediksi pola dari model berdasarkan data yang ada.

Data mining merupakan langkah pada *Knowledge Discovery in Database* (KDD) yang terdiri dari penggunaan algoritma serta penerapan analisis data untuk menghasilkan daftar pola, model atau informasi terhadap data. Tahapan tersebut bersifat interaktif yaitu pengguna terlibat langsung atau dengan perantaraan berbasis pengetahuan.

Knowledge discovery in database (KDD) sering digunakan untuk menjelaskan proses penggalian informasi dari sebuah data yang besar. Istilah tersebut memiliki konsep yang berbeda namun saling berkaitan antara satu dengan lainnya. *Data mining* merupakan salah satu tahapan dari keseluruhan proses KDD.

2.5 *Modified K-Nearest Neighbor (MK-NN)*

Modified K-Nearest Neighbor (MK-NN) adalah metode klasifikasi baru yang dikembangkan dari metode klasifikasi *K-nearest neighbor* (K-NN). Jika K-NN mengklasifikasikan data pengujian berdasarkan nilai tertinggi dari beberapa kelas pada K data pelatihan dengan jarak terdekat, maka MK-NN mengklasifikasikan data pengujian berdasarkan bobot tertinggi dari beberapa kelas pada K data pelatihan yang tervalidasi dengan jarak terdekat (Parvin, 2008).

Validitas dapat memberikan informasi yang lebih banyak tentang keadaan data latih pada fitur dan label kelas dari masing-masing data pelatihan. MK-NN memberikan kesempatan yang lebih besar kepada data latih yang memiliki validitas yang lebih tinggi dan memiliki jarak terdekat dengan data uji, sehingga klasifikasi kelas pada data uji tidak terlalu terpengaruh terhadap data yang tidak konsisten (Parvin, 2008). Dengan adanya validasi pada data latih, MK-NN dapat mengklasifikasikan data uji dengan lebih baik.

Modified K-Nearest Neighbor (MK-NN) terdiri dari dua tahapan. Tahapan pertama yang dilakukan adalah menghitung validitas data latih. Tahapan kedua adalah mengklasifikasikan data pengujian dengan menggunakan *weight voting* dan validitas dari data latih yang didapat sebelumnya.

2.5.2 *Euclidean Distance*

Jarak *Euclidean* paling sering digunakan untuk menghitung jarak. Jarak *euclidean* berfungsi untuk menguji ukuran yang dapat digunakan sebagai kedekatan jarak antara dua objek. Jarak *euclidean* direpresentasikan sebagai berikut (Parvin, 2008):

$$\text{Jarak} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \quad \ldots\ldots (\mathbf{2.7})$$

Keterangan :

Jarak : Jarak dari data p ke q

pi : elemen ke-i dari data p

qi : elemen ke-i dari data q

n : jumlah elemen dari data p dan data q

2.5.3 Validitas Data

Validitas data digunakan untuk menghitung jumlah titik dengan kelas yang sama untuk semua data pada data latih. Validitas data dari tiap data latih tergantung dari data latih lain yang menjadi tetangganya. Setelah validasi data, selanjutnya data tersebut digunakan sebagai informasi lebih mengenai data tersebut (Parvin, 2008). Persamaan yang digunakan untuk menghitung validitas setiap data latih adalah:

$$\text{Validity}\ (x) = \frac{1}{k} \sum_{i=1}^{k} S(lbl(N_i(x))) \quad \ldots\ldots (\mathbf{2.8})$$

Keterangan :

Validity(x) : validitas data ke-x

K : jumlah data tetangga terdekat

lbl(x) : kelas dari data latih ke-x

lbl(Ni(x)) : kelas dari data latih terdekat dari x

S : fungsi similaritas data

Fungsi similaritas *S* digunakan untuk menghitung kesamaan antara titik *a* dan data ke-*b* yang merupakan tetangga terdekat.

$$S(a,b) = \begin{cases} 1 & a=b \\ 0 & a \neq b \end{cases}$$

Keterangan :

a : kelas *a* pada data latih

b : kelas lain selain *a* pada data latih

2.5.4 *Weight Voting*

Dalam metode MK-NN, *weight* dari masing-masing tetangga dihitung dengan menggunaan 1/(de + 0.5). Kemudian, validitas dari setiap data pada data latih dikalikan dengan weight berdasarkan pada jarak euclidean. Sehingga dalam metode MK-NN, didapatkan persamaan *weight voting* untuk setiap tetangga, yaitu sebagai berikut (Parvin, 2008):

$$W(x) = Validity(x)x \frac{1}{d_x+0,5} \quad \text{(2.9)}$$

Keterangan :

W(x) : bobot dari data latih ke-x

Validity(x) : validitas dari data latih ke-x

dx : jarak dari data uji ke data latih x

2.6 Normalisasi

Normalisasi merupakan tahap merubah nilai menjadi kisaran 0 dan 1 (Budianita, 2013). Normalisasi ini merupakan proses penskalaan nilai atribut dari data sehingga bisa jatuh pada *range* tertentu.

Pada perhitungan jarak *Euclidean*, atribut berskala panjang dapat mempunyai pengaruh lebih besar dari pada atribut berskala pendek. Oleh karena itu, untuk mencegah hal tersebut perlu dilakukan normalisasi terhadap nilai atribut. Metode yang dipakai pada penelitian ini adalah metode *Min-Max*, *Min-Max* adalah metode normalisasi dengan melakukan transformasi *linier* terhadap data asli. Persamaan untuk normalisasi atribut X adalah sebagai berikut:

$$X^* = \frac{X - \min(X)}{\max(X) - \min(X)} \quad \text{(2.10)}$$

Keterangan:

X* adalah nilai setelah dinormalisasi,

X adalah nilai sebelum dinormalisasi,

min(X) adalah nilai minimum dari fitur,

dan *max*(X) adalah nilai maksimum dari suatu fitur.

Kelebihan dari metode ini adalah keseimbangan nilai perbandingan antar data saat sebelum dan sesudah proses normalisasi. Tidak ada data bias yang dihasilkan oleh metode ini. Kekurangannya adalah jika terdapat data baru, metode ini akan memungkinkan terjebak *out of bound error*. Metode normalisai *Min-Max* digunakan karena selain dalam data di ketahui nilai minimum dan maksimum nya, *min-max* ini sudah banyak digunakan para peneliti lainnya untuk melakukan normalisasi, karena *min-max* tergolong mudah dan hasil yang dapat adalah tidak bias sehingga mempermudah pengerjaan penormalisasian data dan lebih efisien

2.7 Pengujian Hasil

Di dalam penelitian terdapat beberapa pengujian yang dilakukan yang berguna untuk mengetahui hasil akurasi serta hasil pengujian lainnya, pengujian hasil yang dapat dilakukan yaitu dengan mencari hasil pengujian nilai akurasi.

2.7.1 Akurasi

Tingkat keberhasilan sistem dihitung berdasarkan perbandingan jumlah klasifikasi yang sesuai terhadap seluruh data jenis serangan yang di uji. Untuk pengujian akurasi menggunakan *confusion matrix*. Model *confusion matrix* dapat dilihat pada Tabel 2.2 berikut (Han & Kamber, 2011):

Tabel 2.2 ***Confussion Matrix***

		Actual Class		
		Yes	**No**	**Total**
Predicted Class	**Yes**	TP	TN	P
	No	FP	FN	N
	Total	P'	N'	P + N

Keterangan :

- *True Positive* (TP) merupakan jumlah data dengan nilai sebenarnya positif dan nilai prediksi positif.
- *True Negative* (TN) merupakan jumlah data dengan nilai sebenarnya negatif dan nilai prediksi negatif.
- *False Positive* (FP) merupakan jumlah data dengan jumlah nilai sebenarnya negatif dan nilai prediksi positif.
- *False Negative* (FN) merupakan jumlah data dengan nilai sebenarnya positif dan nilai prediksi negatif.

Berdasarkan *confusion matrix* diatas maka untuk menghitung tingkat akurasi dari klasifikasi dapat dihitung dengan menggunakan Persamaan 2.11 sebagai berikut :

$$\boldsymbol{Akurasi = \frac{TP + TN}{P + N}} \quad \textbf{(2.11)}$$

2.8 *Blackbox Testing*

Pengujian *software* sangat diperlukan untuk memastikan *software* atau sistem yang sudah/sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Pengembang atau penguji *software* harus menyiapkan sesi khusus untuk menguji program yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dikoreksi secepatnya (Mustaqbal, Firdaus, and Rahmadi, 2015).

Black Box Testing berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. *Black Box Testing* bukanlah solusi alternatif dari *White Box* Testing tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box Testing*. *Black Box Testing* cenderung untuk menemukan hal-hal berikut (Mustaqbal, Firdaus, and Rahmadi, 2015):

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*)
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).

5. Kesalahan inisialisasi dan terminasi.

2.9 UML (*Unified Modeling Language*)

Unified Modeling Language atau UML merupakan sebuah bahasa pemodelan sistem yang berparadigma berorientasi objek. UML digunakan sebagai penyederhanaan terhadap permasalahan yang luas sehingga akan lebih mudah untuk dipahami (Nugroho, 2010).

2.9.1 Jenis-Jenis UML Diagram

Terdapat beberapa jenis diagram pada UML yang dapat digunakan dalam perencanaan pembuatan sebuah system. Diagram-diagram tersebut memiliki arti dan fungsi yang berbeda namun tetap saling berhubungan antara satu dengan lainnya. Beberapa diagram tersebut adalah *Use Case Diagram*, *Activity Diagram*, *Class Diagram* dan *Deployment Diagram* (Nugroho, 2010).

1. *Use Case Diagram*

   Use case diagram adalah sebuah diagram yang di dalamnya terdapat aktor dan juga memiliki relasi diantaranya. Untuk memahami dan menganalisa kebutuhan sistem pada saat melakukan perancangan, *use case* diagram dapat dikatakan sebagai titik awalnya sehingga beberapa kebutuhan yang diperlukan dapat ditentukan terlebih dahulu. *Use case* diagram menggambarkan secara detail bagaimana sistem memproses sesuatu dan bagaimana aktor dalam menggunakan sistem (Nugroho, 2010).

2. *Activity Diagram*

   Activity diagram merupakan diagram yang digunakan untuk menganalisa tingkah laku *use case* yang lebih detail serta menunjukkan beberapa relasi diantara keduanya dan juga menggambarkan hubungan bisnis antar satu *use case* dengan *use case* lainnya (Nugroho, 2010).

3. *Class diagram*

   Class diagram berfungsi untuk menggambarkan beberapa perbedaan mendasar antara tiap *class*, hubungan dan sub-sistem *class* tersebut (Nugroho, 2010).

4. *Deployment Diagram*

Deployment diagram digunakan untuk menunjukkan tata letak suatu sistem dan memperlihatkan beberapa bagian dari perangkat lunak *(software*) yang berjalan pada perangkat keras (Nugroho, 2010).

2.10 Penelitian Terkait

Dalam melakukan penelitian, dilakukan pengumpulan materi yang bersangkutan dan penelitian terkait dengan dataset NSL-KDD dan metode yang digunakan. Terdapat beberapa penelitian sejak beberapa tahun belakangan diantaranya pada Tabel 2.3 Berikut:

Tabel 2.3 Penelitian Terkait

NO	Peneliti (Tahun)	Judul	Metode Yang Digunakan	Hasil
1	(Kaushik and Deshmukh, 2011)	*Detection of Attacks in an Intrusion Detection System*	- *K-Nearest Neighbor*	Terdapat beberapa pendekatan untuk mendeteksi serangan dalam sistem pendeteksian intruksi, setiap pendekatan memiliki cara masing-masing dan memliki kelebihan serta kekurangan tersendiri. Terdapat beberapa teknik baru yang muncul dan dapat menghapus kekurangan dari sebelumnya.
2	(Garg and Kumar, 2014)	*Combinational Feature Selection Approach for Network Intrusion Detection System*	- *Symmetrical Uncertainty* - *Gain Ratio* - *Boolean AND Operator* - *IBK*	Kombinasi dari *Feature Selection Symmetrical Uncertainty* dan *Gain Ratio* memiliki kinerja yang tinggi terhadap 15 atribut teratas yang digunakan. 15 Atribut yang digunakan kemudian disaring kembali menggunakan pendekatan *Boolean*

NO	Peneliti (Tahun)	Judul	Metode Yang Digunakan	Hasil
				AND Operator hingga mendapatkan hasil yang lebih mengerucut. Hasil akurasi yang diperoleh dengan kombinasi *Feature Selection* ini sebesar 96,04%.
3	(Okfalisa and Mustakim, 2017)	*Comparative Analysis of K-Nearest Neighbor and Modified K-Nearest Neighbor Algorithm for Data Classification*	- *K-Nearest Neighbor* - *Modified K-Nearest Neighbor*	Analisa perbandingan untuk metode *Modified K-Nearest Neighbor* (MK-NN) dan *K-Nearest Neighbor* (KNN) dilakukan untuk mengetahui kemampuan akurasi untuk klasifikasi dari dua algoritma tersebut. Nilai yang diperoleh oleh metode MK-NN sebesar 99,51% telah menunjukkan bahwa metode tersebut jauh lebih unggul dari metode sebelumnya yaitu K-NN yang memiliki akurasi sebesar 94,95%.
4	(Ahmad and Hayat, 2017)	*Intelligent computational model for classification of sub-Golgi protein using oversampling and fisher feature selection methods.*	- *K-Nearest Neighbor*	Penerapan algoritma *feature selection* pada penelitian ini bertujuan untuk mengurangi ruang fitur dan menghilangkan fitur *noisy* dan redundan data yang tidak diperlukan. Dan hasilnya menunjukkan bahwa dengan menerapkan algoritma *feature selection* dengan metode klasifikasi K-NN dapat

NO	Peneliti (Tahun)	Judul	Metode Yang Digunakan	Hasil
				meningkatkan nilai akurasi dan mempersingkat waktu.
5	(Tiwari and Kumar, 2017)	*INTRUSION DETECTION SYSTEM*	- *Intrusion Detection System*	Dengan menerapkan *feature selection*, pada saat mendeteksi intrusi data akan semakin cepat dilakukan, karena algoritma *feature selection* mampu mempersingkat waktu terhadap masalah tersebut.